

---

# Partial VOROS: A Cost-aware Performance Metric for Binary Classifiers with Precision and Capacity Constraints

---

Christopher Ratigan<sup>1</sup>, Kyle Heuton<sup>2</sup>, Carissa Wang<sup>2</sup>, Lenore Cowen<sup>1,2</sup>, Michael C. Hughes<sup>2</sup>

<sup>1</sup>Department of Mathematics, Tufts University, Medford, MA, USA

<sup>2</sup>Department of Computer Science, Tufts University, Medford, MA, USA

## Abstract

The ROC curve is widely used to assess binary classification performance. Yet for some applications such as deterioration alert systems for hospitalized patient monitoring, conventional ROC analysis cannot capture crucial factors that impact deployment, such as enforcing a minimum precision constraint to avoid false alarm fatigue or imposing an upper bound on the number of predicted positives to represent the capacity of hospital staff. The usual area under the curve metric also does not reflect asymmetric costs for false positives and false negatives. In this paper we address all three of these issues. First, we show how the subset of classifiers that meet given precision and capacity constraints can be represented as a feasible region in ROC space. We establish the geometry of this feasible region. We then define the *partial area of lesser classifiers*, a performance metric that is monotonic with cost and only accounts for the feasible portion of ROC space. Averaging this area over a desired range of cost parameters results in the partial volume over the ROC surface, or *partial VOROS*. In experiments predicting mortality risk using vital sign history on the MIMIC-IV dataset, we show this cost-aware metric is better than alternatives for ranking classifiers in hospital alert applications.

## 1 INTRODUCTION

When a classifier of binary events is deployed in a high-stakes application, it must respect important opera-

tional constraints. First, context-specific costs mean that false positive predictions usually have different consequences than false negatives in a given task. Developing and evaluating classifiers in a cost-aware way is key to deployment success (Provost and Fawcett, 1997; Drummond and Holte, 2000). Second, stakeholders may specify viable ranges for certain performance metrics for the system to be beneficial. Finally, stakeholders may have capacity constraints, in terms of the overall number of positive predictions or negative predictions they can handle smoothly when deployed.

In this work, our goal is to develop a performance metric that can effectively rank classifiers when costs, performance constraints, and capacity constraints all matter. Previous work has suggested many metrics and visuals for evaluating binary classifiers, surveyed later in Sec. 7. We take as a starting point a cost-aware analysis of the receiver-operating-characteristic (ROC) curve. Recent work by Ratigan and Cowen (2025) lifts the classic 2D ROC curve into a 3D surface where the third axis defines the cost. They proposed a performance metric, the volume over the ROC surface or *VOROS*, that can identify when a binary classifier outperforms another given a task-specific range of estimated costs. Our work here extends this analysis to incorporate constraints on precision and capacity, which are critical in many applications.

As a motivating task of interest, consider the evaluation of alert systems for monitoring the health of hospitalized patients. In such systems, a classifier must take in recent data about a patient’s health and determine whether or not to alarm. Each alarm indicates the patient’s health may be deteriorating and signals that doctors or nurses should check on that patient soon. An ever-increasing body of literature has made progress on alert systems developed via machine learning (Abella Álvarez et al., 2013; Hyland et al., 2020; Sendak et al., 2020; Muralitharan et al., 2021; Edelson et al., 2024). Careful evaluation of such systems is

critical to ensure they balance tradeoffs appropriately and provide a net benefit to the patient population and the hospital.

Hospital-based alert systems naturally have the three aforementioned operational constraints:

- First, there are asymmetric costs to the different kinds of mistakes. A false positive has some cost by taking valuable time from clinical staff that could be spent on other patients with greater needs. A false negative incurs even more cost, as this means a patient did get sicker or even die and an alert that might have helped was never issued.
- Second, hospital staff often express a key performance metric constraint: the alert classifier’s *precision*, the fraction of all alarms that are true positives, needs to meet some minimum value for the system to be viable (Harrison et al., 2015; Rath and Hughes, 2022). A survey of critical care physicians in South Korean hospitals (Park et al., 2022) found that too many false positives were the top concern about early warning systems; the median response requested a precision of at least 28.5%. Clinical staff may learn to ignore many or all alarms from the system altogether if its precision is not tolerable, a problem known as *alarm fatigue* (Cvach, 2012). Ignoring alarms entirely due to low precision has been a documented safety concern for decades (Sendelbach and Funk, 2013; Albanowski et al., 2023).
- Finally, the staff’s capacity to respond to alarms must be accounted for. In a typical ICU in the U.S., the patient-to-doctor ratio is on average 11.8 and almost always above 6 (Kahn et al., 2023). If alarms for all patients happened at once, only a fraction could be triaged right away. It is crucial to ensure system evaluation takes into account such constraints.

In this paper, we develop a new performance metric, the *Partial VOROS*, which extends the VOROS from Ratigan and Cowen (2025) to account for precision and capacity constraints, while preserving the original VOROS’ ability to handle asymmetric costs. In Sec. 3, we formalize how these constraints narrow the entirety of ROC space to a feasible region satisfying all constraints. In Sec. 4, we provide computable formulas for evaluating areas of lesser classifiers and volumes over this feasible region using desired cost ranges. In Sec. 5 and 6, we examine classifiers on real health records data, showing how our partial VOROS can help identify promising classifiers by accounting for constraints and costs in ways other metrics cannot.

## 2 Background

We consider an observed dataset  $\mathcal{D} = \{X_i, Y_i\}_{i=1}^{N_*}$  of  $N_*$  total pairs indexed by  $i$  of a feature vector  $X_i \in \mathcal{X}$  and

its associated binary label  $Y_i \in \{0, 1\}$ . Let  $\mathcal{P}$  denote the subset of this dataset with positive labels  $Y_i = 1$ , and  $\mathcal{N} = \mathcal{D} \setminus \mathcal{P}$  denote the subset of all negative labels.

We wish to use dataset  $\mathcal{D}$  to evaluate a score-producing binary classifier  $\mathcal{F} : \mathcal{X} \rightarrow \mathbb{R}$ . This score can be thresholded to produce a binary prediction  $\hat{Y}_i$ . We refer to a classifier using threshold  $\tau$  to make binary predictions as a **binarized classifier**, denoted  $\mathcal{F}_\tau$ . By comparing predictions to true labels over all  $N_*$  examples, we can count the number of true positives  $N_{TP}$ , false positives  $N_{FP}$ , true negatives  $N_{TN}$ , and false negatives  $N_{FN}$ .

We now establish concepts needed for a cost-aware ROC analysis, following Ratigan and Cowen (2025).

**Definition 1** (ROC Space). *The performance of binarized classifier  $\mathcal{F}_\tau$  on dataset  $\mathcal{D}$  at a particular threshold  $\tau$  can be represented as a point  $(h, k)$  in two-dimensional space where  $h$  is the false positive rate and  $k$  the true positive rate. We refer to the set of all possible such  $(h, k)$  points, which span the unit square  $[0, 1] \times [0, 1]$ , as **ROC space**.*

ROC space is useful for assessing classifier performance with respect to cost. Let  $C_0$  define the cost of a false positive and  $C_1$  the cost of a false negative. Given a dataset, the total cost of all possible mistakes is given by  $C_0|\mathcal{N}| + C_1|\mathcal{P}|$ . Naturally, assigning costs to points in ROC space depends on the relative sizes of  $C_0$  and  $C_1$  and the relative sizes of  $|\mathcal{P}|$  and  $|\mathcal{N}|$ . Following Ratigan and Cowen (2025), we can capture this dependency in one parameter  $t \in [0.0, 1.0]$ .

**Definition 2** (Fractional Cost Parameter). *Let  $t = \frac{C_0|\mathcal{N}|}{C_0|\mathcal{N}| + C_1|\mathcal{P}|}$  denote the portion of aggregate misclassification cost due to false positives. We have  $0 \leq t \leq 1$ .*

In general for a fixed  $\mathcal{D}$ , larger values of  $t$  imply the cost of a false positive is larger:  $C_0 > C_1$ . We cannot compare  $t$  across datasets with different class balances.

**Definition 3** (Cost). *The **normalized cost** of a ROC point  $(h, k)$  for data  $\mathcal{D}$  given cost parameter  $t$  is*

$$Cost_t(h, k) = th + (1 - t)(1 - k) = \frac{C_0 N_{FP} + C_1 N_{TP}}{C_0 |\mathcal{N}| + C_1 |\mathcal{P}|}$$

The worst possible point in ROC space,  $(1, 0)$ , would have cost of 1.0. The best point  $(0, 1)$  would have cost 0.0. All other points have cost in between 0.0 and 1.0, reflecting their cost relative to the worst point.

This formulation of cost as a linear function of ROC coordinates  $(h, k)$  is well-known (Drummond and Holte, 2000). However, Ratigan and Cowen (2025) was the first paper to add a separate axis based on the  $t$  parameter defined above to ROC space to compare classifiers

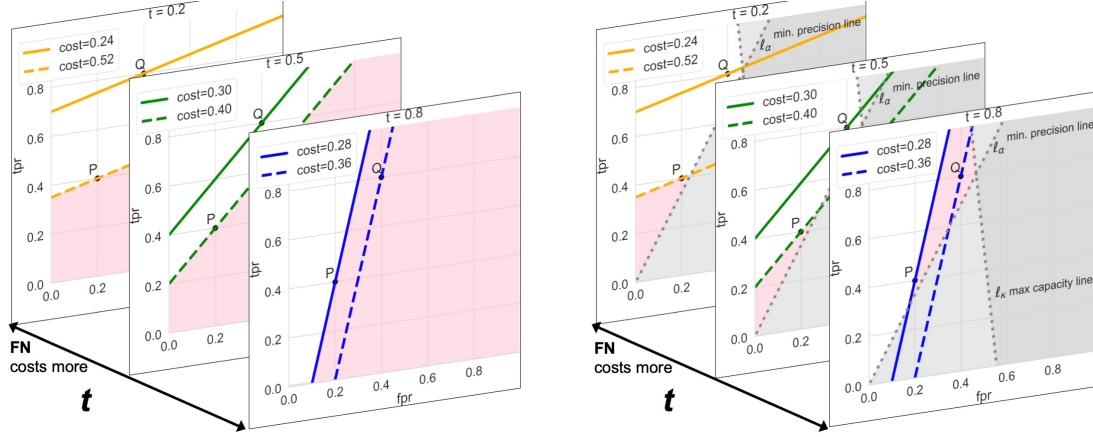


Figure 1: Overview of VOROS (left) and our new partial VOROS (right). *Left*: At each fractional cost parameter  $t \in [0, 1]$ , we draw iso-performance lines for points  $P$  and  $Q$  in ROC space. Solid lines have lower cost than dashed lines. Points below each line have higher cost than that line. A point’s *area of lesser classifiers*, colored light pink here for point  $P$ , is the area below that point’s iso-performance line. The VOROS (Ratigan and Cowen, 2025) for a classifier is computed by finding at each  $t$  the maximum area of lesser classifiers for any point in its ROC curve, then integrating over a desired range of  $t$ . *Right*: Our partial VOROS **excludes** regions in gray that do not achieve a minimum precision  $\alpha$  or exceed a maximum capacity  $\kappa$  (yield too many positive predictions). These limits correspond to linear constraints in ROC space.

with respect to variable costs. The key notion here was the ROC surface defined below.

**Definition 4** (ROC Surface). *Let  $(h, k)$  be a point in ROC space, the **ROC surface** associated to  $(h, k)$  is the saddle surface in 3D space with coordinates  $x, y, t$  given equivalently by  $t = \frac{y-k}{y-k+x-h}$  and  $y = \frac{t}{1-t}(x-h) + k$ , where  $t$  is the fractional cost from Def. 2.*

Given a fixed  $t$  value, the second formulation here represents all points  $(x, y)$  in ROC space with the same cost as the point  $(h, k)$ .

This same-cost set is known as an *iso-performance line* (Provost and Fawcett, 2001).

**Definition 5** (Iso-performance line). *Let  $(h, k)$  be a point in ROC space and let  $t$  be a fixed fractional cost parameter. Then the line in ROC space*

$$y = \frac{t}{1-t}(x-h) + k$$

*represents all points  $(x, y)$  with the same cost as  $(h, k)$  using  $t$  and is called an **iso-performance line**.*

Each ROC panel in Fig. 1 visualizes for a specific  $t$  the iso-performance lines for the same two points  $P$  and  $Q$ . Varying  $t$  adjusts the slope of the iso-performance line.

### 3 Bounds on ROC Space

A common criticism of ROC space and the area under the ROC curve is that it fails to measure the performance of a binary classifier under appropriate operational constraints. These criticisms still hold for the

ROC surface and the volume over the ROC surface measure of Ratigan and Cowen (2025). In this section we introduce two conditions – a bound on precision and a bound on capacity – that restrict the allowed classifiers in ROC space. These constraints are motivated by early warning classifiers for hospitalized patient monitoring (Harrison et al., 2015; Rath and Hughes, 2022). We then define a portion of ROC space meeting these constraints and other key assumptions.

#### 3.1 Precision Bound

**Definition 6** (Precision). *The precision of the binarized classifier  $\mathcal{F}_\tau$  on dataset  $\mathcal{D}$  is the fraction of positive predictions that are correct:*

$$Prec = \frac{N_{TP}}{N_{TP} + N_{FP}}$$

Precision is also known as positive predictive value.

**Definition 7** (Precision Bound). *To be feasible, a classifier must satisfy a **minimum precision bound**:*

$$Prec \geq \alpha$$

Here, the desired precision  $\alpha > 0$  can be set by talking with stakeholders about their tolerance for false alarms. This constraint is motivated by applications of ML to early warning alert systems, especially in hospitals (Harrison et al., 2015; Rath and Hughes, 2022).

For dataset  $\mathcal{D}$ , let  $p = \frac{|\mathcal{P}|}{|\mathcal{D}|}$  define the **prevalence**, the fraction of all examples that are positive. Using  $p$ , then we can recast the definition of precision itself, as well as the bound above, in terms of ROC coordinates.

**Lemma 8** (Precision Bound in ROC Space). *Let  $\mathcal{F}_\tau$  be a binarized classifier with ROC coordinates  $(x, y)$  on a dataset with positive prevalence  $p$ . Then its precision is at least  $\alpha$  iff*

$$y \geq \frac{\alpha(1-p)}{(1-\alpha)p}x$$

*We call the line in ROC space where this holds with equality the **minimum precision line**  $\ell_\alpha$ .*

*Proof.* Starting from the original precision bound, we write precision in terms of  $x, y$  and  $p$ , using  $N_{TP} = py$  and  $N_{FP} = (1-p)x$ . Then, solve for  $y$  in terms of  $x$ :

$$\frac{py}{py + (1-p)x} \geq \alpha \Rightarrow y \geq \frac{\alpha(1-p)}{(1-\alpha)p}x$$

This algebra is valid when  $1-\alpha > 0, p > 0$ .  $\square$

This inequality means that if we require a priori that classifiers have at least  $\alpha$  precision, then rather than the entire unit square of ROC space, we only consider the region above the minimum precision line  $\ell_\alpha : y = \frac{\alpha(1-p)}{(1-\alpha)p}x$  through the origin. When  $p$  is small or when  $\alpha$  is large, then we ignore a large portion of ROC space.

### 3.2 Capacity Bound

Beyond precision, another issue prevalent in applications is that the system that responds to alerts has a maximal *capacity* for positive predictions. Unlike precision, which is a rate, capacity is an absolute (though sometimes soft) cutoff due to limited resources. In hospital alert systems, capacity constraints arise due to limited time to tend to patients by existing staff. In information retrieval, there may be limited time to handle relevant documents. Throughout, we assume that *not alarming* has no impact on capacity. This is reasonable in hospitals, as no resources beyond standard care need be allocated when there is no alarm.

**Definition 9** (Capacity bound). *To be feasible, the total number of predicted positives a classifier produces must not exceed a provided **maximum capacity**.*

$$N_{TP} + N_{FP} \leq \kappa$$

Here, the value of  $\kappa > 0$  can be set by stakeholders, indicating the maximum number of alerts that can be handled by the system given typical resources.

**Lemma 10** (Capacity Bound in ROC space). *Let  $\mathcal{F}_\tau$  be a binarized classifier with ROC coordinates  $(x, y)$  for dataset  $\mathcal{D}$ . If capacity bound  $\kappa$  is satisfied, then*

1. *The number of predicted positives is:  $|\mathcal{P}|y + |\mathcal{N}|x$ .*
2. *We have  $y \leq \frac{\kappa - |\mathcal{N}|x}{|\mathcal{P}|}$ .*

We call the line in ROC space where this holds with equality the **maximum capacity line**,  $\ell_\kappa$ .

*Proof.* The number of predicted positives is  $N_{TP} + N_{FP} = |\mathcal{P}|y + |\mathcal{N}|x$  by definition. Bound satisfaction means we have  $|\mathcal{P}|y + |\mathcal{N}|x \leq \kappa$ . Solving for  $y$  yields the desired inequality.  $\square$

### 3.3 Feasible Region

We now consider enforcing both precision and capacity constraints, along with some practical assumptions typical in our target applications. These constraints narrow down ROC space to a particular smaller region we call the feasible region.

**Definition 11** (Practical Assumptions). *From here on, we assume our dataset and bound limits satisfy:*

- $|\mathcal{P}| < |\mathcal{N}|$ : *Negatives are more common.*
- $p < \alpha < 1.0$ : *Precision is reasonable.*
- $0 < \kappa < |\mathcal{D}|$ : *Capacity is non-trivial.*
- $t < \frac{\alpha N}{\alpha N + (1-\alpha)P}$ : *Never-alarm has maximal cost.*

Note  $\kappa < |\mathcal{D}|$  ensures that the always-alarm baseline is not feasible. The  $t$  bound in the last item ensures that the never-alarm baseline maximizes cost for the feasible region (see appendix).

**Definition 12** (Feasible classifier). *Let  $\mathcal{F}_\tau$  be a binarized classifier with ROC coordinates  $(h, k)$ . Given a dataset, precision limit  $\alpha$  and capacity limit  $\kappa$  that meet Def. 11, we call  $\mathcal{F}_\tau$  a **feasible classifier** and  $(h, k)$  a **feasible point** if  $Prec \geq \alpha$  and  $N_{TP} + N_{FP} \leq \kappa$ .*

**Definition 13** (Feasible Region). *Let  $S$  be a set of constraints on classifiers for dataset  $\mathcal{D}$ . Then, the **feasible region** in ROC space is the set of all  $(h, k)$  points in ROC space satisfying all constraints in  $S$ .*

This is similar to what Morasca and Lavazza (2020) call a “region of interest”, though ours incorporates precision and capacity.

### 3.4 Geometry of the Feasible Region

We can define the geometry of the feasible region via the minimum precision line  $\ell_\alpha$  and the maximum capacity line  $\ell_\kappa$ . The line  $\ell_\alpha$  has its y-intercept at the origin, and slope within  $(1.0, +\infty)$  for  $\alpha \in (p, 1.0)$  by Def. 11. Similarly,  $\ell_\kappa$  has fixed slope  $-\frac{|\mathcal{N}|}{|\mathcal{P}|}$  (always  $< -1$ ), with free y-intercept  $\frac{\kappa}{|\mathcal{P}|}$  for  $\kappa \in (0, |\mathcal{D}|)$ . Where these lines intersect determines what kind of polygon the feasible region forms. The 3 cases below exhaustively cover all possible  $\kappa$  for a given feasible value of  $\alpha \in (p, 1.0)$  subject to the practical assumptions made in Definition 11, as diagrammed in Figure 2. Edge cases that

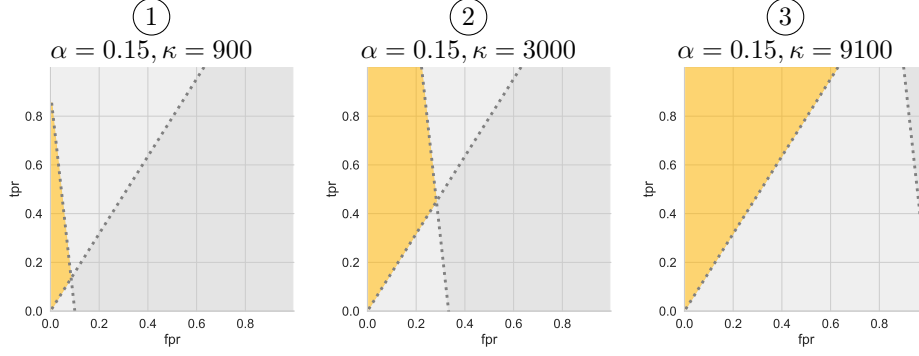


Figure 2: Cases for distinct polygons of feasible region (colored in gold), assuming the practical facts in Def. 11. Created using  $|\mathcal{P}| = 1000$ ,  $|\mathcal{N}| = 9000$ , so  $p = 0.1$ , roughly matching the mortality alert application on MIMIC-IV in Sec. 6.

violate the assumptions of Definition 11 are handled in the appendix.

- ①  $0 < \kappa < |\mathcal{P}|$ : Triangle that excludes (0,1). Here,  $\ell_\alpha$  and  $\ell_\kappa$  intersect inside ROC space. The capacity bound eliminates the perfect classifier.
- ②  $|\mathcal{P}| \leq \kappa < \frac{1}{\alpha}|\mathcal{P}|$ . Quadrilateral including (0,1).  $\ell_\alpha$  and  $\ell_\kappa$  intersect inside ROC space, with capacity allowing the perfect classifier.
- ③  $\frac{1}{\alpha}|\mathcal{P}| \leq \kappa < |\mathcal{D}|$ . Right Triangle including (0,1).  $\ell_\alpha$  and  $\ell_\kappa$  intersect above ROC space (above the  $y=1$  upper edge), so the capacity constraint is made ineffective and precision alone dominates.

Below, we precisely define the polygon enclosing the feasible region in terms of specific vertices for each case.

**Definition 14** (Notation for vertices). Let  $i, j \in \{0, 1\}$  and let  $\beta \in \{\alpha, \kappa\}$ . Define the following 9 vertices for all possible  $i, j, \beta$

- (1-4)  $v_{ij}$  is the intersection of  $x = i$  and  $y = j$
- (5-6)  $v_{\beta j}$  is the intersection of  $\ell_\beta$  and  $y = j$
- (7-8)  $v_{i\beta}$  is the intersection of  $x = i$  and  $\ell_\beta$ .
- (9)  $v_{\alpha\kappa}$  is the intersection of  $\ell_\alpha$  and  $\ell_\kappa$ .

**Definition 15** (Feasible Region Polygon). Clockwise from the origin, the bounding vertices of the region of interest are  $v_{00}v_{0\kappa}v_{\alpha\kappa}$  in case 1,  $v_{00}v_{01}v_{\kappa1}v_{\alpha\kappa}$  in case 2, and  $v_{00}v_{01}v_{\alpha1}$  in case 3.

**Area of Feasible Region.** Given the vertices defining the feasible region, we can compute its area by applying the well-known “shoelace” formula (Lee and Lim, 2017; Zwillinger, 2018), which computes the area of a polygon given its boundary vertices (see appendix).

## 4 Partial Area and Volume

With the geometry of the feasible region established, we now consider how to perform *cost-aware* ranking of classifiers in this region. Ratigan and Cowen (2025) introduced two key ideas for cost-aware ranking in

their unconstrained setting: the notion of a lesser classifier and the area of lesser classifiers in ROC space. In Sec. 4.1, we extend these ideas to enforce a minimum precision  $\alpha$  and a maximum capacity  $\kappa$ . Next, in Sec. 4.2, we then explain how to lift this area analysis to 3D space  $(x, y, t)$ , and how to integrate over a given  $t$  range to compute a partial volume.

### 4.1 Partial Area

**Definition 16** (Lesser classifier). Let  $\mathcal{F}_1, \mathcal{F}_2$  be feasible binarized classifiers with ROC coordinates  $(h_1, k_1), (h_2, k_2)$  on dataset  $\mathcal{D}$ . Then  $\mathcal{F}_1$  is a **lesser classifier** of  $\mathcal{F}_2$  at cost parameter  $t$  if it has worse cost:  $\text{Cost}_t(h_1, k_1) > \text{Cost}_t(h_2, k_2)$ .

**Definition 17** (Partial area of lesser classifiers). Let  $\mathcal{F}_\tau$  be a feasible classifier and let  $t$  be a fractional cost parameter. The **partial area of lesser classifiers**,  $A_t^*(\mathcal{F}_\tau)$  is the area of the portion of the feasible region of ROC space consisting of  $\mathcal{F}_\tau$ ’s lesser feasible classifiers using cost parameter  $t$ .

By Definition 16 the partial area of lesser classifiers is cost monotone. Under some values of  $\alpha$  and  $\kappa$ , feasible classifiers may occupy only a small region of ROC space. The numerical value of the partial area may thus be small even for the best feasible classifiers. For a metric whose value is easy to interpret as good or bad regardless of  $\alpha, \kappa$ , we recommend a normalization:

**Definition 18** (Normalized partial area). Let  $\mathcal{F}_\tau$  be a feasible classifier and let  $t$  be a fractional cost parameter. The **normalized partial area** of lesser classifiers is the ratio of the partial area of lesser classifiers  $A_t^*(\mathcal{F}_\tau)$  to the partial area of all feasible points in ROC space.

We can compute the denominator in this ratio directly using the area of the feasible region (see appendix). If the perfect classifier located at (0, 1) in ROC space is feasible, then this denominator is *equivalent* to the area of lesser classifiers for the perfect classifier.

$$A_t^*(h, k) = \begin{cases} \frac{1}{2} \left( A'(h+k) + \left( \frac{B'h}{C'-D'} - A'h \right) \frac{1}{1-t} \left( \frac{B'C'h}{C'-D'} - B'(h+k) \right) \frac{1}{C't-D} \right) \\ \frac{1}{2} \left( \frac{(1-\alpha)\kappa(\kappa-|\mathcal{N}|A)}{|\mathcal{N}||\mathcal{P}|} - \frac{A\alpha\kappa}{|\mathcal{P}|} + A(h+k) - \left( Ah + \frac{Bh}{C+D} \right) \frac{1}{1-t} + \left( B(h+k) - \frac{\kappa B}{|\mathcal{P}|} - \frac{CBh}{C+D} \right) \frac{1}{Ct+D} \right) \\ \frac{1}{2} \left( \frac{(1-\alpha)\kappa}{|\mathcal{N}|} - \frac{\alpha\kappa^2 - \alpha\kappa|\mathcal{P}|}{|\mathcal{P}||\mathcal{N}|} + \frac{\kappa - |\mathcal{P}|}{|\mathcal{N}|} + (k+h-1)^2 - \frac{h^2}{1-t} - \frac{(1-k)^2}{t} \right) \\ \frac{1}{2} \left( \frac{(1-\alpha)|\mathcal{P}|}{\alpha|\mathcal{N}|} - \frac{h^2}{1-t} - \frac{(1-k)^2}{t} + h^2 + k^2 - 2(h+k) + 2hk + 1 \right) \end{cases}$$

Figure 3: Formula for partial area of lesser classifiers for a given  $(h, k)$  location in ROC space and cost parameter  $t \in [0, 1]$ . The 4 cases here depend on the geometry of the feasible region and the iso-performance line  $\ell_t$  through  $(h, k)$ . The appendix identifies which case is needed for given inputs. The cases above are, in order, the areas of  $v_{00}v_{\alpha t}v_{0t}$ ,  $v_{00}v_{\alpha\kappa}v_{\kappa t}v_{0t}$ ,  $v_{00}v_{\alpha\kappa}v_{\kappa 1}v_{t1}v_{0t}$ , and  $v_{00}v_{\alpha 1}v_{t1}v_{0t}$  respectively, using the vertex definitions in Def. 14.

Our notion of normalized partial area of lesser classifiers is similar in spirit to the ratio of relevant areas defined by Morasca and Lavazza (2020).

**Lemma 19** (Form of the Partial Area). *The partial area of lesser classifiers is a rational linear function of  $t$  as given in Figure 3.*

*Proof.* A full proof is in the appendix. The area depends on how the iso-performance line  $\ell_t$  through  $(h, k)$  intersects the feasible region. The key insight is that the  $x$ -coordinate of  $\ell_t \cap \ell_\alpha$  can be expressed as  $A' - \frac{B'}{C't - D'}$  and that of  $\ell_t \cap \ell_\kappa$  as  $A + \frac{B}{Ct + D}$ , for suitable scalars  $A, B, C, D, A', B', C', D'$ .  $\square$

## 4.2 Partial Volume over ROC surface

When stakeholders have a range of cost parameters  $t$  in mind as well as many possible thresholds  $\tau$ , Ratigan and Cowen (2025) showed how integrating the area of lesser classifiers over that range of  $t$  leads to a performance metric called the volume over the ROC surface (VOROS). Importantly, at each  $t$ , they sensibly use the largest area over all thresholds  $\tau$ . We now extend this idea to a *partial volume* that only considers the feasible region imposed by precision and capacity constraints.

**Definition 20** (Partial VOROS). *Let  $\mathcal{F}$  be a score-producing classifier, let  $[a, b] \subseteq [0, 1]$  be a cost parameter range, and let  $\alpha, \kappa$  define precision and capacity limits so that the feasible region is well-defined with area  $A^*$ . Then the **partial volume over the ROC surface (VOROS)** is the normalized partial area of lesser classifiers, averaged over the provided range:*

$$PV(\mathcal{F}) = \frac{1}{(b-a)A^*} \int_{t=a}^b \max_{\tau} (A_t^*(\mathcal{F}_\tau)) dt.$$

Here, the maximum is taken over the set of thresholds for  $\mathcal{F}$  that produce distinct feasible points  $(h, k)$ .

The maximum in the definition is computable in  $O(h_c)$ -time from the convex hull of an ROC curve where  $h_c$  is the number of feasible points in the convex hull. See appendix for an algorithm. Because  $A_t^*$  is monotonically decreasing in cost, finding the minimum cost of all  $h_c$  points via Eq. (3) is faster than assessing area at each point; only one area calculation is needed per  $t$ .

Overall, partial VOROS (PV) is a higher-is-better performance metric with interpretable values between 0.0 and 1.0 regardless of the dataset or constraints  $\alpha, \kappa$ . Given the practical assumptions of Definition 11, the best possible PV is 1.0, achieved by the highest feasible classifier on the  $y$ -axis. The worst possible PV is 0.0, achieved by the baseline of never alarming.

Similar to the VOROS and traditional Area under the ROC curve, the partial VOROS will always assign a higher value to a curve that dominates.

**Lemma 21.** *Let  $y = f_1(x)$ ,  $y = f_2(x)$  be two ROC curves. If for all  $x$ ,  $f_1(x) \geq f_2(x)$ , then the VOROS and partial VOROS of  $f_1$  will be higher than  $f_2$ .*

*Proof.* Since the curve for  $f_1(x)$  is above and to the left of the curve  $f_2(x)$ , for any fractional cost parameter  $t$ , the best performing point on  $f_2(x)$  will be in the area of lesser classifiers of some point on  $f_1(x)$ .  $\square$

## 5 Evaluating Off-the-Shelf EW Scores

Here, we look at partial VOROS as a post-hoc metric for existing clinical early warning (EW) scores. We re-analyze ROC curves published by Edelson et al. (2024), measuring the performance of widely-used risk scores on 362,926 patient stays in 7 hospitals in Connecticut, USA. The outcome of interest is deterioration, meaning transfer to the ICU or death within 24 hours. The prevalence  $p$  is 4.6% in Edelson et al.’s data.

For this case study, we focus on 3 scoring methods: NEWS (Royal College of Physicians, 2012), NEWS2 (Royal College of Physicians, 2017), and the

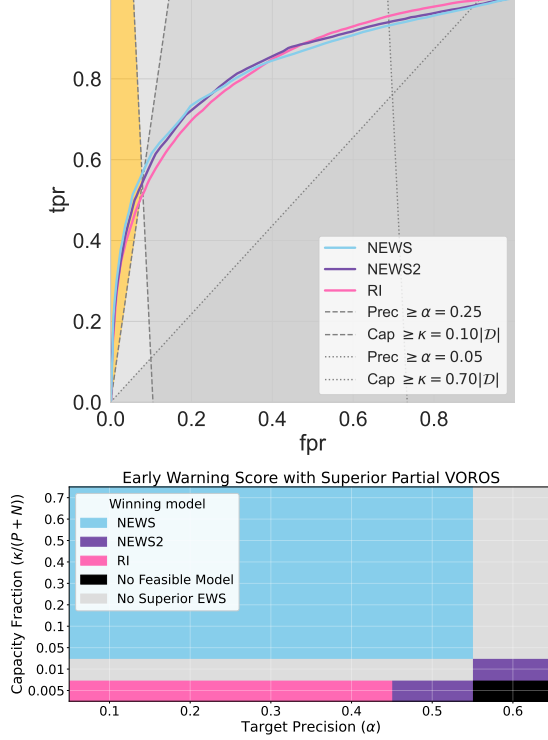


Figure 4: **Re-analysis of Early Warning Scores for Deterioration**, using ROC curves from Edelson et al. (2024). Top: ROC curves with strict (– lines) and more lenient (·: ) constraints. Bottom: Heatmap of models with superior partial VOROS scores over possible  $\alpha, \kappa$  limits.

Rothman Index (RI, Rothman et al. (2013)). We visualize ROC curves in Fig. 4: each curve crosses others, and it is thus imperative to consider costs and constraints to determine the best operating point of all methods. We consider capacity  $\kappa$  from 0.5% – 70% of  $|\mathcal{D}|$  and precision limits  $\alpha \in (0.1, 0.6)$ . We examine cost parameters  $t \in (0.01, 0.7)$ , so that Def. 11 is satisfied.

Over the range of  $\alpha, \kappa$  limits, we plot a heatmap indicating where different models have clear wins and regions where pairs of models are indistinguishable (normalized PV is within 0.01). We find that NEWS2 preforms best at high target precision and low capacity, RI has superior recall at lower target precision and low capacity, and NEWS performs best once capacity is greater than 5% of all examples. Edelson et al. (2024) previously concluded that NEWS “outperformed” the two other scores here; our analysis with partial VOROS adds nuance. Each method could be the clear winner depending on precision and capacity constraints.

## 6 Developing a Mortality Alert System

Our aim here is to illustrate how different performance metrics lead to different selections for classifier-and-threshold on a realistic alert deployment task. We’ll

show how cost-aware and cost-unaware selection strategies perform in terms of ultimate cost on test data. We focus on in-hospital mortality prediction, inspired by evaluations in Rath and Hughes (2022) but using the updated MIMIC-IV dataset (Johnson et al., 2023). We aim for competitive *reproducible* classifiers, not state-of-the-art on past cost-unaware benchmarks.

**Task description.** For each ICU patient-stay, we observe basic facts at admission (age, insurance type, weight, gender) as well as time-varying health signals over the first 48 hours (6 vitals and 7 labs, details in Supplement), selected to follow Rath and Hughes. The prediction task is to identify if the patient will die during the rest of their hospital stay. We use an open-source pipeline (Gupta et al., 2022) to obtain a cohort of viable patient-stays, each as a 0-48 hr time series with values every 2 hour window. We keep only stays produced by Gupta et al.’s code with at least 3 channels (vitals or labs) measured in the last 16 hours.

Our train/valid./test sets contain 15474/7802/7861 patient-stays with prevalence  $p$  of .104/.103/.107. This split was done by patient-id in a label-stratified way. We favor larger test sets than usual to be sure we can measure precision well despite low prevalence.

**Featurization.** For each of 13 time-varying univariate channels, we extract features that represent 7 summary functions (was ever measured, time since last measurement, mean, variance, min, median, max, slope) over 3 windows (0-48 hours, 24-48 hours, 32-48 hours). Missing values in extracted features are filled with the population mean, then rescaled to the 0.0-1.0 range.

**Classifiers.** We examine logistic regression, multi-layer perceptron, and random forests, using sklearn (Pedregosa et al., 2011). For each, we conduct an extensive grid search of 48+ hyperparameter configurations to avoid overfitting (details in supplement).

**Cost ranges.** We define our cost parameter intervals in terms of  $\frac{C_0}{C_1}$ , then map to  $t$  via Def. 2. We consider two cost-constraint scenarios. First, we fix  $\alpha = 0.15, \kappa = 0.5|\mathcal{D}|$ , and use  $\frac{C_0}{C_1} \sim \text{Unif}(\frac{1}{9}, \frac{1}{6})$ , which leads to a non-uniform density for  $t \in (0.5, 0.6)$  when computing the partial VOROS. Here, false negatives are more costly than false positives, but by less than 10x. Second, we choose a more constrained example, setting  $\alpha = 0.5, \kappa = 0.1|\mathcal{D}|$  to reflect the clinical priorities of high precision and low total alarms. Recent economic estimates (Rogers et al., 2023) suggest a missed clinical deterioration may have roughly 20-40 times the cost of a false alarm, so we use  $\frac{C_0}{C_1} \sim \text{Unif}(0.025, 0.05)$ , which is a non-uniform density for  $t \in (.18, .31)$ .

**Evaluation Plan.** We compare 4 possible selection strategies: max pAUROC and max recall in feasible

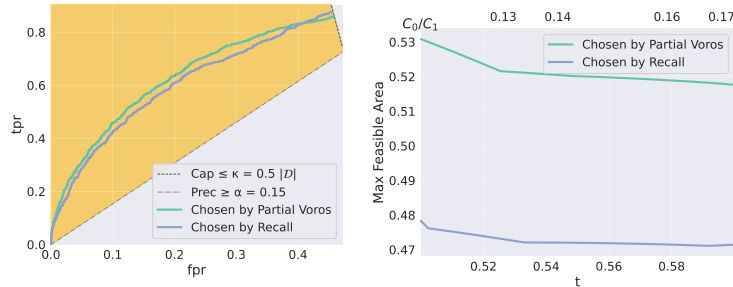


Figure 5: **Developing mortality alert systems on MIMIC-IV.** We compare our *partial VOROS* against other strategies for selecting model family, hyperparameters, and decision thresholds using validation data. We evaluate each selection’s binary alerts on test data in terms of average cost over the provided  $C_0/C_1$  distribution. *Left:* The ROC curve on heldout data in the feasible region (gold), for the two best selection strategies in scenario 1. *Center:* The maximum feasible area of lesser classifiers across a range of cost parameter values  $t$ , with the corresponding cost ratio  $C_0/C_1$  (false positive to false negative) on the top axis. *Right:* Table of costs on test set for each strategy and scenario.

( $\alpha, \kappa$ -aware;  $t$ -unaware), max VOROS ( $\alpha, \kappa$ -unaware;  $t$ -aware), and max PV (aware of both). Using each one, we search the hyperparameter grid of over 250 ROC curves on validation data. For each possible strategy, we select one curve, and then a threshold,  $\tau$ , to use at each  $t$ . We then evaluate on test data each of the model-threshold pairs selected on validation. This forces us to see how actual binary alert systems behave on new data; we don’t get to post-hoc select thresholds on test.

**Results.** Fig. 5(a) shows two ROC curves selected by different strategies; Fig. 5(b) shows their respective normalized partial areas as a function of  $t$ . Evaluation of expected cost on test for each strategy is in Fig. 5(c), where the table reports a Monte Carlo average over  $C_0/C_1$  samples each mapped to a  $t$ .

**Analysis.** In scenario 1 ( $t \in [0.5, 0.6]$ ), our max partial VOROS (PV) strategy yields better (lower) normalized cost on test data than alternatives. Scenario 2 favors the lower left of ROC curves where there is less crossing; we find most strategies sensibly have similar costs, except Ratigan and Cowen’s VOROS is worse.

## 7 Related Work

For an overview of performance metrics see Hand (2012); for a focus on visuals, see (Prati et al., 2011). Steyerberg and Vergouwe (2014) provides advice for healthcare focused modeling, where beyond just assessing discrimination via ROC, calibration is also valuable.

A common summary of the ROC is the area under the curve (Bradley, 1997; Hand, 2012), known as AUROC or the concordance statistic. Many works recommend a *partial area* under the ROC by only integrating over some false positive rates (McClish, 1989; Jiang et al., 1996; Robin et al., 2011). Our partial VOROS can be seen as a 3D extension of the partial AUROC that

focuses on a desired cost range and obeys both capacity and precision constraints. Other alternative partial area metrics seek better correspondence to concordance (Carrington et al., 2020) but do not account for capacity or precision.

Shao et al. (2024) extend ROC curves to cost-sensitive learning, seeking to make weighted area-under-curve training robust to train-to-test shifts in cost and class distribution. Their work does not address precision or capacity constraints, unlike our partial VOROS.

Particularly for hospital alert systems, some works recommend *precision-recall* curves instead of (or in addition to) ROC curves (Saito and Rehmsmeier, 2015; Romero-Brufau et al., 2015; Martin et al., 2025). However, claims that the PR curve or the area under it (AUPRC) is somehow superior to the ROC/AUROC for imbalanced data have been recently refuted (Richardson et al., 2024; McDermott et al., 2024). For more on PR curves, see Flach and Kull (2015).

### Training models to meet operational constraints.

Synergistically with our work on a new performance metrics, other work has sought to train models directly to perform well on certain metrics (Tsoi et al., 2022; Eban et al., 2017). Some of these directly optimize for recall at a precision constraint (Rath and Hughes, 2022; Fathony and Kolter, 2020; Peng et al., 2025) or the area under the PR curve (Ramzi et al., 2021).

## 8 Conclusion

We have developed the *partial volume-over-the-ROC surface* as a performance metric for classifiers that accounts for cost-imbalance, precision constraints, and capacity constraints. Our work represents careful geometric reasoning about tradeoffs between recall, precision, and false alarm rates. Our experiments on real

---

health records show how partial VOROS can help make model selection decisions that would meet necessary operational constraints and reduce costs when deployed.

**Limitations.** We focus on binary classification; future work would be needed to transfer these ideas to a many-class or multi-label setting. Our work requires numerical costs and constraints to be specified. It may be challenging to elicit values for technical constraints from non-technical stakeholders, though some work suggests routes forward (Wu et al., 2008).

## Acknowledgements

The authors acknowledge support from the U.S. National Science Foundation (NSF) via two awards: IIS #2338962 and the DIAMONDS REU #2149871.

## References

- Abella Álvarez, A., Torrejón Pérez, I., Enciso Calderón, V., Hermosa Gelbard, C., Sicilia Urban, J. J., Ruiz Grinspan, M., García Ureña, M. Á., Salinas Gabiña, I., Mozo Martín, T., Calvo Herranz, E., Díaz Blázquez, M., and Gordo Vidal, F. (2013). ICU without walls project. Effect of the early detection of patients at risk. *Medicina Intensiva*, 37(1):12–18.
- Albanowski, K., Burdick, K. J., Bonafide, C. P., Kleinpell, R., and Schlesinger, J. J. (2023). Ten Years Later, Alarm Fatigue Is Still a Safety Concern. *AACN Advanced Critical Care*, 34(3):189–197.
- Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7).
- Carrington, A. M., Fieguth, P. W., Qazi, H., Holzinger, A., Chen, H. H., Mayr, F., and Manuel, D. G. (2020). A new concordant partial AUC and partial c statistic for imbalanced data in the evaluation of machine learning algorithms. *BMC Medical Informatics and Decision Making*, 20(1):4.
- Chan, T. M. (1996). Optimal output-sensitive convex hull algorithms in two and three dimensions. *Discrete & computational geometry*, 16(4):361–368.
- Cvach, M. (2012). Monitor Alarm Fatigue: An Integrative Review. *Biomedical Instrumentation & Technology*, 46(4):268–277.
- Drummond, C. and Holte, R. C. (2000). Explicitly representing expected cost: An alternative to ROC representation. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 198–207.
- Eban, E., Schain, M., Mackey, A., Gordon, A., Rifkin, R., and Elidan, G. (2017). Scalable Learning of Non-Decomposable Objectives. In *Artificial Intelligence and Statistics*, pages 832–840.
- Edelson, D. P., Churpek, M. M., Carey, K. A., Lin, Z., Huang, C., Siner, J. M., Johnson, J., Krumholz, H. M., and Rhodes, D. J. (2024). Early Warning Scores With and Without Artificial Intelligence. *JAMA Network Open*, 7(10):e2438986.
- Fathony, R. and Kolter, Z. (2020). AP-perf: Incorporating generic performance metrics in differentiable learning. In *International Conference on Artificial Intelligence and Statistics*, pages 4130–4140. PMLR.
- Flach, P. and Kull, M. (2015). Precision-Recall-Gain Curves: PR Analysis Done Right. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.
- Gupta, M., Gallamozza, B., Cutrona, N., Dhakal, P., Poulain, R., and Beheshti, R. (2022). An extensive data processing pipeline for {MIMIC-IV}. In *Machine Learning for Health (ML4H)*. PMLR.
- Hand, D. J. (2012). Assessing the performance of classification methods. *International Statistical Review*, 80(3):400–414.
- Harrison, A. M., Herasevich, V., and Gajic, O. (2015). Automated Sepsis Detection, Alert, and Clinical Decision Support: Act on It or Silence the Alarm? *Critical Care Medicine*, 43(8):1776.
- Hyland, S. L., Faltys, M., Hüser, M., Lyu, X., Gumbsch, T., Esteban, C., Bock, C., Horn, M., Moor, M., Rieck, B., Zimmermann, M., Bodenham, D., Borgwardt, K., Rättsch, G., and Merz, T. M. (2020). Early prediction of circulatory failure in the intensive care unit using machine learning. *Nature Medicine*, 26(3):364–373.
- Jiang, Y., Metz, C. E., and Nishikawa, R. M. (1996). A receiver operating characteristic partial area index for highly sensitive diagnostic tests. *Radiology*, 201(3):745–750.
- Johnson, A. E., Bulgarelli, L., Shen, L., Gayles, A., Shammout, A., Horng, S., Pollard, T. J., Hao, S., Moody, B., Gow, B., et al. (2023). MIMIC-IV, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1.
- Kahn, J. M., Yabes, J. G., Bukowski, L. A., and Davis, B. S. (2023). Intensivist physician-to-patient ratios and mortality in the intensive care unit. *Intensive Care Medicine*, 49(5):545–553.
- Lee, Y. and Lim, W. (2017). Shoelace Formula: Connecting the Area of a Polygon and the Vector Cross Product. *The Mathematics Teacher*, 110(8):631–636.
- Martin, B., Bennett, T. D., DeWitt, P. E., Russell, S., and Sanchez-Pinto, L. N. (2025). Use of the Area Under the Precision-Recall Curve to Evaluate Prediction Models of Rare Critical Illness Events. *Pediatric Critical Care Medicine*, 26(6):e855–e859.

- 
- McClish, D. K. (1989). Analyzing a portion of the ROC curve. *Medical Decision Making: An International Journal of the Society for Medical Decision Making*, 9(3):190–195.
- McDermott, M. B., Zhang, H., Hansen, L. H., Angelotti, G., and Gallifant, J. (2024). A closer look at AUROC and AUPRC under class imbalance. In Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., and Zhang, C., editors, *Advances in Neural Information Processing Systems*, volume 37, pages 44102–44163. Curran Associates, Inc.
- Morasca, S. and Lavazza, L. (2020). On the assessment of software defect prediction models via ROC curves. *Empirical Software Engineering*, 25(5).
- Muralitharan, S., Nelson, W., Di, S., McGillion, M., Devereaux, P. J., Barr, N. G., and Petch, J. (2021). Machine Learning–Based Early Warning Systems for Clinical Deterioration: Systematic Scoping Review. *Journal of Medical Internet Research*, 23(2):e25187.
- Park, S.-H., Kang, J., Kim, T. J., Lee, H. Y., Cho, H.-J., and Lee, S.-M. (2022). Current status of the rapid response system and early warning score: A survey-based analysis. *Acute and Critical Care*, 37(4):687–689.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Peng, L., Travadi, Y., He, C., Cui, Y., and Sun, J. (2025). Exact Reformulation and Optimization for Direct Metric Optimization in Binary Imbalanced Classification.
- Prati, R. C., Batista, G. E. A. P. A., and Monard, M. C. (2011). A Survey on Graphical Methods for Classification Predictive Performance Evaluation. *IEEE Transactions on Knowledge and Data Engineering*, 23(11):1601–1618.
- Provost, F. and Fawcett, T. (1997). Analysis and visualization of classifier performance with nonuniform class and cost distributions. In *Proceedings of AAAI-97 Workshop on AI Approaches to Fraud Detection & Risk Management*, pages 57–63.
- Provost, F. and Fawcett, T. (2001). Robust classification for imprecise environments. *Machine learning*, 42:203–231.
- Ramzi, E., THOME, N., Rambour, C., Audebert, N., and Bitot, X. (2021). Robust and decomposable average precision for image retrieval. In *Advances in Neural Information Processing Systems*.
- Rath, P. and Hughes, M. (2022). Optimizing early warning classifiers to control false alarms via a minimum precision constraint. In *International Conference on Artificial Intelligence and Statistics*, pages 4895–4914. PMLR.
- Ratigan, C. and Cowen, L. (2025). The VOROS: Lifting ROC curves to 3D to summarize unbalanced classifier performance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 20148–20156.
- Richardson, E., Trevizani, R., Greenbaum, J. A., Carter, H., Nielsen, M., and Peters, B. (2024). The receiver operating characteristic curve accurately assesses imbalanced datasets. *Patterns*, 5(6).
- Robin, X., Turck, N., Hainard, A., Tiberti, N., Lisacek, F., Sanchez, J.-C., and Müller, M. (2011). pROC: An open-source package for R and S+ to analyze and compare ROC curves. *BMC bioinformatics*, 12:77.
- Rogers, P., Boussina, A. E., Shashikumar, S. P., Wardi, G., Longhurst, C. A., and Nemati, S. (2023). Optimizing the Implementation of Clinical Predictive Models to Minimize National Costs: Sepsis Case Study. *Journal of Medical Internet Research*, 25:e43486.
- Romero-Brufau, S., Huddleston, J. M., Escobar, G. J., and Liebow, M. (2015). Why the C-statistic is not informative to evaluate early warning scores and what metrics to use. *Critical Care*, 19(1).
- Rothman, M. J., Rothman, S. I., and Beals, J. (2013). Development and validation of a continuous measure of patient condition using the Electronic Medical Record. *Journal of Biomedical Informatics*, 46(5):837–848.
- Royal College of Physicians (2012). National Early Warning Score ({NEWS}): Standardising the assessment of acute-illness severity in the {NHS}. Report of a working party, {RCP}, London.
- Royal College of Physicians (2017). National Early Warning Score ({NEWS}) 2: Standardising the assessment of acute-illness severity in the {NHS}. Updated report of a working party, RCP, London.
- Saito, T. and Rehmsmeier, M. (2015). The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PLoS ONE*, 10(3).
- Sendak, M. P., Ratliff, W., Sarro, D., Alderton, E., Futoma, J., Gao, M., Nichols, M., Revoir, M., Yashar, F., Miller, C., Kester, K., Sandhu, S., Corey, K., Brajer, N., Tan, C., Lin, A., Brown, T., Engelbosch, S., Anstrom, K., Elish, M. C., Heller, K., Donohoe, R., Theiling, J., Poon, E., Balu, S., Bedoya, A., and O’Brien, C. (2020). Real-World Integration of

- 
- a Sepsis Deep Learning Technology Into Routine Clinical Care: Implementation Study. *JMIR Medical Informatics*, 8(7).
- Sendelbach, S. and Funk, M. (2013). Alarm fatigue: A patient safety concern. *AACN advanced critical care*, 24(4):378–386.
- Shao, H., Xu, Q., Yang, Z., Wen, P., Peifeng, G., and Huang, Q. (2024). Weighted ROC curve in cost space: Extending AUC to cost-sensitive learning. *Advances in Neural Information Processing Systems*, 36.
- Steyerberg, E. W. and Vergouwe, Y. (2014). Towards better clinical prediction models: Seven steps for development and an ABCD for validation. *European Heart Journal*, 35(29):1925–1931.
- Tsoi, N., Candon, K., Li, D., Milkessa, Y., and Vázquez, M. (2022). Bridging the gap: Unifying the training and evaluation of neural network binary classifiers. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pages 23121–23134. Curran Associates, Inc.
- Wang, S., McDermott, M. B. A., Chauhan, G., Ghassemi, M., Hughes, M. C., and Naumann, T. (2020). MIMIC-Extract: A data extraction, preprocessing, and representation pipeline for MIMIC-III. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, CHIL ’20, pages 222–235, New York, NY, USA. Association for Computing Machinery.
- Wu, T., Huang, H., Du, G., and Sun, Y. (2008). A novel partial area index of receiver operating characteristic (ROC) curve. In *Medical Imaging 2008: Image Perception, Observer Performance, and Technology Assessment*, volume 6917, pages 82–89. SPIE.
- Zwillinger, D. (2018). Ch. 4.7: Polygons. In *CRC Standard Mathematical Tables and Formulas*, Advances in Applied Mathematics, page 212. CRC Press, 33 edition.

---

## A Cases for Feasible Region Polygons

In this section, we provide a detailed analysis of the feasible region as well as explicit formulas for the vertices and areas of the feasible region.

Using the notation of Definition 14, we have the following formulas for the potential vertices of the feasible region.

**Lemma 22** (Coordinates of Vertices). *If  $i, j \in \{0, 1\}$ , then  $v_{ij} = (i, j)$ , also if  $i = 0$ , then  $v_{i\alpha} = v_{\alpha i} = (0, 0)$ . Otherwise, we have*

- $v_{\alpha\kappa} = \left(\frac{(1-\alpha)\kappa}{|\mathcal{N}|}, \frac{\alpha\kappa}{|\mathcal{P}|}\right)$
- $v_{\alpha 1} = \left(\frac{(1-\alpha)|\mathcal{P}|}{\alpha|\mathcal{N}|}, 1\right)$
- $v_{\kappa 1} = \left(\frac{\kappa - |\mathcal{P}|}{|\mathcal{N}|}, 1\right)$
- $v_{0\kappa} = \left(0, \frac{\kappa}{|\mathcal{P}|}\right)$
- $v_{\kappa 0} = \left(\frac{\kappa}{|\mathcal{N}|}, 0\right)$
- $v_{1\kappa} = \left(1, \frac{\kappa - |\mathcal{N}|}{|\mathcal{P}|}\right)$
- $v_{1\alpha} = \left(1, \frac{\alpha|\mathcal{N}|}{(1-\alpha)|\mathcal{P}|}\right)$

This lemma follows directly from simple algebra using the definition.

In this section, we explain how to handle cases that don't fit into Definition 11.

Note: since  $\ell_\alpha$  passes through the origin, we have  $v_{00} = v_{\alpha 0} = v_{0\alpha} = (0, 0)$  for all  $\alpha$ , so there are really only 7 points of interest for our analysis.

With a precision  $\alpha \geq 0$  and maximum capacity  $\kappa \geq 0$ , there are 13 cases for our feasible region, we break these into 8 degenerate and 5 nondegenerate cases, where the 3 cases that satisfy Definition 11 are 3 of the 5 non-degenerate cases, and the other two non-degenerate cases violate Definition 11 because  $\alpha < p$ . Let  $\ell_\alpha$  denote the precision line and  $\ell_\kappa$  denote the capacity line.

### A.1 Degenerate Cases

First, note that if  $\alpha = 1$  or  $\kappa = 0$ , then the feasible region is merely the point  $(0, 1)$  or  $(0, 0)$  respectively. Also, the feasible region is empty if  $\alpha = 1$  and  $\kappa \leq |\mathcal{D}|$ .

If  $\alpha = 0$  or  $\kappa \geq |\mathcal{D}|$ , we consider the feasible region to be degenerate as only one of the two bounds intersects the interior of ROC space.

The degenerate cases depend on how  $\ell_\alpha$  or  $\ell_\kappa$  intersect ROC space. If all we have is a minimum precision bound, there are two possibilities.  $v_{\alpha 1}$  borders ROC space, or  $v_{1\alpha}$  borders ROC space. Note the first case is equivalent to nondegenerate case 1 below. Similarly, if all we have is a capacity bound, there are four cases depending on which pair of  $v_{\kappa 0}, v_{\kappa 1}, v_{0\kappa}$  and  $v_{1\kappa}$  border ROC space (note  $v_{0\kappa}$  and  $v_{\kappa 1}$  cannot both border ROC space since  $\ell_\kappa$  has nonpositive slope).

### A.2 Nondegenerate cases

Assume that  $\alpha \in (0, 1)$  and  $\kappa \in (0, |\mathcal{D}|)$ , then we have five cases.

Case 1:  $v_{\alpha\kappa}$  lies inside of ROC space.

This also splits into 2 subcases depending on whether  $v_{01}$  is feasible according to  $\ell_\kappa$ .

Case 1A: If  $v_{01}$  is not feasible then the feasible region is the triangle  $\triangle v_{00}v_{\alpha\kappa}v_{0\kappa}$ . This is Case 1 in the main paper.

Case 1B: If  $v_{01}$  is feasible, then the feasible region is the quadrilateral  $v_{00}v_{\alpha\kappa}v_{\kappa 1}v_{01}$ . This is Case 2 in the main paper.

Case 2:  $v_{\alpha\kappa}$  lies above  $y = 1$ . (This is Case 3 in the main paper)

In this case, the feasible region is simply the triangle  $\triangle v_{00}v_{\alpha 1}v_{01}$ .

Case 3:  $v_{\alpha\kappa}$  lies to the right of  $x = 1$ . (Not in the main paper, violates Definition 11 since  $\alpha < p$ )

This splits into 2 subcases depending on whether  $v_{01}$  is feasible according to  $\ell_\kappa$ .

Case 3A: if  $v_{01}$  is feasible, then the feasible region is the pentagon  $v_{00}, v_{1\alpha}v_{1\kappa}v_{\kappa 1}v_{01}$

Case 3B: if  $v_{01}$  is not feasible, then the feasible region is the trapezoid  $v_{00}v_{1\alpha}v_{1\kappa}v_{0\kappa}$ .

### A.3 Details on Cases Satisfying Definition 11

In this section we detail the form and area of the feasible region subject to the cases in Section 3.4 (and also denoted Cases A, C1 and C2 above):

**Lemma 23** (Vertices for feasible regions). *Counterclockwise from the origin, the bounding vertices of the Case 1 Triangle are*

- $v_{00} = (0, 0)$
- $v_{\alpha\kappa} = \left(\frac{(1-\alpha)\kappa}{|\mathcal{N}|}, \frac{\alpha\kappa}{|\mathcal{P}|}\right)$
- $v_{0\kappa} = \left(0, \frac{\kappa}{|\mathcal{P}|}\right)$

*Counterclockwise from the origin, the bounding vertices of the Case 2 Quadrilateral are*

1.  $v_{00} = (0, 0)$
2.  $v_{\alpha\kappa} = \left(\frac{(1-\alpha)\kappa}{|\mathcal{N}|}, \frac{\alpha\kappa}{|\mathcal{P}|}\right)$
3.  $v_{\kappa 1} = \left(\frac{\kappa - |\mathcal{P}|}{|\mathcal{N}|}, 1\right)$
4.  $v_{01} = (0, 1)$

*Counterclockwise from the origin, the bounding vertices of the Case 3 triangle are*

1.  $v_{00} = (0, 0)$
2.  $v_{\alpha 1} = \left(\frac{(1-\alpha)|\mathcal{P}|}{\alpha|\mathcal{N}|}, 1\right)$
3.  $v_{01} = (0, 1)$

Each of these vertices can be found by simple algebra.

In order to decide which of the 3 cases to use in the area formula given by 29, we can use the following lemma.

**Lemma 24** (Area of Feasible Region). *In general, the area of the feasible region in ROC space is*

$$A^* = \begin{cases} \frac{(1-\alpha)\kappa^2}{2|\mathcal{N}||\mathcal{P}|} & \text{case 1} \\ \frac{2\kappa|\mathcal{P}| - \alpha\kappa^2 - |\mathcal{P}|^2}{2|\mathcal{N}||\mathcal{P}|} & \text{case 2} \\ \frac{(1-\alpha)|\mathcal{P}|}{\alpha|\mathcal{N}|} & \text{case 3} \end{cases}$$

*Proof.* This follows by applying the well-known “shoelace” formula (Lee and Lim, 2017; Zwillinger, 2018) for the area of a polygon applied to the boundary vertices defined above.  $\square$

This lemma shows that the area of the feasible region can be calculated precisely from the constraints defining the problem and dataset. Alternatively, since the feasible region consists of the convex hull of 6 lines in the plane, software can calculate the area of the convex hull of the collection of 6 lines  $x = 0$ ,  $x = 1$ ,  $y = 0$ ,  $y = 1$ ,  $\ell_\kappa$ , and  $\ell_\alpha$ , quite quickly (see Section C below).

---

## B Cases for Partial Areas

Given that we are in one of the three cases from Section 3.4, there are four cases for the region of lesser classifiers.

First, some notation

**Definition 25.** *Let  $i \in \{0, 1, \alpha, \kappa\}$ , then*

- $v_{0t}$  is the intersection of  $x = 0$  and  $\ell_t$ .
- $v_{t1}$  is the intersection of  $y = 1$  and  $\ell_t$ .
- $v_{\alpha t}$  is the intersection of  $\ell_\alpha$  and  $\ell_t$ .
- $v_{\kappa t}$  is the intersection of  $\ell_\kappa$  and  $\ell_t$ .

Assuming the practical assumptions in Definition 11 we have the following Lemma.

**Lemma 26.** *The isoperformance line through the baseline of never alarming does not intersect the interior of the feasible region iff  $t \leq \frac{\alpha|\mathcal{N}|}{\alpha|\mathcal{N}| + (1-\alpha)|\mathcal{P}|}$ .*

*Proof.* The isoperformance line through  $(0,0)$  is  $y = \frac{t}{1-t}x$  which lies below  $\ell_\alpha$  in the first quadrant precisely when  $t < \frac{\alpha|\mathcal{N}|}{\alpha|\mathcal{N}| + (1-\alpha)|\mathcal{P}|}$ . If equality holds, the lines coincide, but  $\ell_t$  still misses the interior.  $\square$

This lemma means that the baseline of never alarming has the highest cost of any point in our feasible region, it also ensures that the partial VOROS of this baseline is 0.

**Lemma 27** (Coordinates for vertices.). *We have the following*

- $v_{0t} = (0, k - \frac{t}{1-t}h)$ .
- $v_{t1} = (\frac{(1-t)(1-k)}{t} + h, 1)$
- $v_{\alpha t}$  has  $x$ -coordinate given by  $A' - \frac{B'}{C't - D'}$  where
  - $A' = \frac{(1-\alpha)|\mathcal{P}|(h+k)}{\alpha|\mathcal{N}| + (1-\alpha)|\mathcal{P}|}$
  - $B' = (1-\alpha)|\mathcal{P}| \left( k - \frac{(h+k)\alpha|\mathcal{N}|}{\alpha|\mathcal{N}| + (1-\alpha)|\mathcal{P}|} \right)$
  - $C' = \alpha|\mathcal{N}| + (1-\alpha)|\mathcal{P}|$
  - $D' = \alpha|\mathcal{N}|$
- $v_{\kappa t}$  has  $x$ -coordinate  $A + \frac{B}{Ct + D}$  where
  - $A = \frac{|\mathcal{P}|(h+k) - \kappa}{|\mathcal{P}| - |\mathcal{N}|}$
  - $B = \kappa - k|\mathcal{P}| - \frac{|\mathcal{N}|(|\mathcal{P}|(h+k) - \kappa)}{|\mathcal{P}| - |\mathcal{N}|}$
  - $C = |\mathcal{P}| - |\mathcal{N}|$
  - $D = |\mathcal{N}|$

*Proof.* The proof follows from taking the intersection of the lines  $\ell_t$ ,  $\ell_\alpha$ ,  $\ell_\kappa$ ,  $y = 1$  and  $x = 0$  as needed. The rational linear forms for the  $x$ -coordinates of  $v_{\alpha t}$  and  $v_{\kappa t}$  follow from the fact that the coefficient of  $x$  in  $\ell_t$  is rational in  $t$ .  $\square$

**Lemma 28.** *The region of lesser classifiers for a feasible  $(h, k)$  is always one of:*

- The triangle  $v_{00}v_{\alpha t}v_{0t}$ .
- The quadrilateral  $v_{00}v_{\alpha\kappa}v_{\kappa t}v_{0t}$
- The pentagon  $v_{00}v_{\alpha\kappa}v_{\kappa 1}v_{t1}v_{0t}$ .
- The quadrilateral  $v_{00}v_{\alpha 1}v_{t1}v_{0t}$ .

Where, the triangle  $v_{00}v_{\alpha t}v_{0t}$  can apply to any of the three cases in section 3.4, so long as  $\ell_t$  intersects  $\ell_\alpha$  below  $\ell_\kappa$  and below  $y = 1$ . The quadrilateral  $v_{00}v_{\alpha\kappa}v_{\kappa t}v_{0t}$ , can only apply to cases 1 and 2 from section 3.4 since  $\ell_t$  needs to intersect  $\ell_\kappa$  on the border of the feasible region. The pentagon only applies when  $\ell_t$  intersects  $y = 1$  on the border of the feasible region in case 2 from section 3.4. Finally, the quadrilateral  $v_{00}v_{\alpha 1}v_{t1}v_{0t}$  only applies when  $\ell_t$  intersects  $y = 1$  on the border of the feasible region in case 3.

*Proof.* Since we are in one of the three cases from Section 3.4, note that the isoperformance line will always leave the feasible region on the left through the  $y$ -axis by the last part of Definition 11, so  $v_{0t}$  is a vertex of all the regions.

The remaining possible regions depend on how the line leaves to the right and which points are included in the polygon. Specifically,

1. if we are in Case 1, the region of lesser classifiers is the triangle  $v_{00}v_{\alpha t}v_{0t}$  if  $\ell_t$  lies on or below  $v_{\alpha\kappa}$ , and is the quadrilateral  $v_{00}v_{\alpha\kappa}v_{\kappa t}v_{0t}$  otherwise.
2. in Case 2, the region of lesser classifiers is the triangle  $v_{00}v_{\alpha t}v_{0t}$  if  $\ell_t$  lies on or below  $v_{\alpha\kappa}$ , the quadrilateral  $v_{00}v_{\alpha\kappa}v_{\kappa t}v_{0t}$  if  $\ell_t$  lies between  $v_{\alpha\kappa}$  and  $v_{\kappa 1}$ , or the pentagon  $v_{00}v_{\alpha\kappa}v_{\kappa 1}v_{t1}v_{0t}$  if  $\ell_t$  lies above  $v_{\kappa 1}$ .
3. in Case 3, the region of lesser classifiers is the triangle  $v_{00}v_{\alpha t}v_{0t}$  if  $\ell_t$  lies on or below  $v_{\alpha 1}$ , or the quadrilateral  $v_{00}v_{\alpha 1}v_{t1}v_{0t}$  otherwise.

□

It is worth noting that whether  $\ell_t$  lies above or below any of these vertices can be readily checked by comparing the cost of  $(h, k)$  to that of the boundary point (higher cost points lie below lower costs ones), so this combined with Lemma 23 yields a simple algorithm for determining the form of the partial area.

**Lemma 29** (Calculating Partial Area). *Let  $t \in [0, 1]$  be fixed, and let  $(h, k)$  be the ROC coordinates of an allowed classifier, then the (non-normalized) partial area of lesser classifiers is given by the formula in Figure 3.*

The formula for the first case relies on the  $x$  coordinate of  $v_{\alpha t}$ ,  $x = A' - \frac{B'}{C't - D'}$ , where we define

$$\begin{aligned} A' &= \frac{(1 - \alpha)|\mathcal{P}|(h + k)}{\alpha|\mathcal{N}| + (1 - \alpha)|\mathcal{P}|} \\ B' &= (1 - \alpha)|\mathcal{P}| \left( k - \frac{(h + k)\alpha|\mathcal{N}|}{\alpha|\mathcal{N}| + (1 - \alpha)|\mathcal{P}|} \right) \\ C' &= \alpha|\mathcal{N}| + (1 - \alpha)|\mathcal{P}| \\ D' &= \alpha|\mathcal{N}|. \end{aligned}$$

The formula of the second case relies on the  $x$  coordinate of  $v_{\kappa t}$ ,  $x = A + \frac{B}{Ct + D}$ , where

$$\begin{aligned} A &= \frac{|\mathcal{P}|(h + k) - \kappa}{|\mathcal{P}| - |\mathcal{N}|} \\ B &= \kappa - k|\mathcal{P}| - \frac{(|\mathcal{P}|(h + k) - \kappa)|\mathcal{N}|}{|\mathcal{P}| - |\mathcal{N}|} \\ C &= |\mathcal{P}| - |\mathcal{N}| \\ D &= |\mathcal{N}| \end{aligned}$$

*Proof.* The proof follows directly from applying the well-known determinant formula for a polygon. Each formula corresponds to a distinct polygon from the preceding lemma. Specifically:

- In the first formula, we are calculating the area of triangle  $v_{00}v_{\alpha t}v_{0t}$ .
- In the second formula, we are calculating the area of the quadrilateral  $v_{00}v_{\alpha\kappa}v_{\kappa t}v_{0t}$ .
- In the third formula, we are calculating the area of the pentagon,  $v_{00}v_{\alpha\kappa}v_{\kappa 1}v_{t1}v_{0t}$ .
- In the fourth formula we are calculating the area of the quadrilateral  $v_{00}v_{\alpha 1}v_{t1}v_{0t}$

□

---

## C Computational Complexity

Given a classifier  $\mathcal{F}$ , we can produce an ROC curve by taking all possible binarized classifiers  $\mathcal{F}_\tau$  and plotting them in ROC space. To distinguish which  $\mathcal{F}_\tau$  are potentially useful, we can take the convex hull of the curve together with the point  $(1, 0)$  in  $O(n \log(h))$  time, where  $h$  is the number of vertices in the Convex Hull using Chan’s Algorithm [Chan \(1996\)](#).

Then, given the convex hull, we can associate to each feasible point a range of values of  $t$  for which that point will have the lowest cost on the curve and hence the highest area of lesser classifiers.

Let  $\{(x_i, y_i)\}$  the points of the convex hull oriented clockwise from  $(0, 0)$  to  $(1, 1)$

**if**  $(x_{i+1}, y_{i+1})$  is feasible **then**

**if**  $x_i = 0$  **then**

**if**  $x_{i+1} = 0$  **then**

            Skip.

**else**

$(x_i, y_i)$  is optimal for  $t \in [\frac{y_{i+1}-y_i}{x_{i+1}-x_i+y_{i+1}-y_i}, 1]$

**end if**

**else**

**if**  $y_i = 1$  **then**

$(x_i, y_i)$  is optimal for  $t$  in  $[0, \frac{y_i-y_{i-1}}{x_i-x_{i-1}+y_i-y_{i-1}}]$ ; Break

**else**

$(x_i, y_i)$  is optimal for  $t$  in  $[\frac{y_{i+1}-y_i}{x_{i+1}-x_i+y_{i+1}-y_i}, \frac{y_i-y_{i-1}}{x_i-x_{i-1}+y_i-y_{i-1}}]$

**end if**

**end if**

**else**

**if**  $(x_i, y_i)$  is feasible. **then**

$(x_i, y_i)$  is optimal for  $t \in [0, \frac{y_i-y_{i-1}}{x_i-x_{i-1}+y_i-y_{i-1}}]$

        Break

**end if**

**end if**

The only computations in this algorithm are the values of  $\frac{y_{i+1}-y_i}{x_{i+1}-x_i+y_{i+1}-y_i}$  for all but the last feasible  $(x_i, y_i)$  and checks whether each  $(x_i, y_i)$  is feasible. Together, this takes order  $n_{\alpha_K}$  time where  $n_{\alpha_K}$  is the number of feasible points on the convex hull.

From here, we can use the formula for partial area in Figure 3 to calculate the areas and average them in  $O(n_{\alpha_K})$  time.

Since the formulas in Figure 3 are rational linear in  $t$ , we can also integrate them directly getting a formula for the Volume associated to each  $t$  range involving a logarithm. Computing this is again  $O(n_{\alpha_K})$ .

This yields an overall computational complexity of  $O(n_{\alpha_K} \log n_{\alpha_K})$  if we need to compute the ROC convex hull or  $O(n_{\alpha_K})$  if we already have the convex hull to begin with.

## D Details on Experiments

### D.1 License and availability

Our experiments in Sec. 6 use the MIMIC-IV dataset ([Johnson et al., 2023](#)), which is freely available to qualified researchers subject to the PhysioNet Credentialed Health Data License 1.5.0.

### D.2 MIMIC-IV: Features used

For the mortality prediction experiments in the main paper, we examine 6 vital signs, collected via bedside monitors and extracted from health records via the CHARTEVENTS table in MIMIC-IV. We further examine 7 laboratory measurements from extracted blood and other fluids, again from the CHARTEVENTS table in MIMIC-IV. See listing in Tab. 1

These vitals and labs are extracted via best practices in *clinical grouping* of conceptually similar variables that

| VITALS                     |                   | LAB MEASUREMENTS       |                        |
|----------------------------|-------------------|------------------------|------------------------|
| Vital Sign                 | feat_name in code | Lab Measurement        | feat_name in code      |
| Blood pressure (diastolic) | bp_diastolic_mmHg | Cholesterol            | cholesterol            |
| Blood pressure (systolic)  | bp_systolic_mmHg  | Glucose                | glucose                |
| Heart rate                 | heart_rate        | Hemoglobin             | hemoglobin             |
| Oxygen saturation          | oxygen_saturation | Lactic Acid            | lactic_acid            |
| Respiratory Rate           | resp_rate         | pH                     | pH                     |
| Temperature                | temp              | platelets              | platelets              |
|                            |                   | white blood cell count | white_blood_cell_count |

Table 1: Summary of 6 vitals and 7 labs used in MIMIC-IV experiments

| itemid | label                                 | unitname | feat_name              |
|--------|---------------------------------------|----------|------------------------|
| 224643 | Manual Blood Pressure Diastolic Left  | mmHg     | bp_diastolic_mmHg      |
| 225310 | ART BP Diastolic                      | mmHg     | bp_diastolic_mmHg      |
| 220180 | Non Invasive Blood Pressure diastolic | mmHg     | bp_diastolic_mmHg      |
| 220051 | Arterial Blood Pressure diastolic     | mmHg     | bp_diastolic_mmHg      |
| 227243 | Manual Blood Pressure Systolic Right  | mmHg     | bp_systolic_mmHg       |
| 224167 | Manual Blood Pressure Systolic Left   | mmHg     | bp_systolic_mmHg       |
| 220179 | Non Invasive Blood Pressure systolic  | mmHg     | bp_systolic_mmHg       |
| 225309 | ART BP Systolic                       | mmHg     | bp_systolic_mmHg       |
| 220050 | Arterial Blood Pressure systolic      | mmHg     | bp_systolic_mmHg       |
| 220603 | Cholesterol                           |          | cholesterol            |
| 220621 | Glucose (serum)                       |          | glucose                |
| 226537 | Glucose (whole blood)                 |          | glucose                |
| 220045 | Heart Rate                            | bpm      | heart_rate             |
| 226730 | Height (cm)                           | cm       | height                 |
| 226707 | Height                                | Inch     | height                 |
| 220228 | Hemoglobin                            | g/dl     | hemoglobin             |
| 225668 | Lactic Acid                           |          | lactic_acid            |
| 220277 | O2 saturation pulseoxymetry           | %        | oxygen_saturation      |
| 220227 | Arterial O2 Saturation                | %        | oxygen_saturation      |
| 223830 | PH (Arterial)                         |          | pH                     |
| 220274 | PH (Venous)                           |          | pH                     |
| 227457 | Platelet Count                        |          | platelets              |
| 224422 | Spont RR                              | bpm      | resp_rate              |
| 220210 | Respiratory Rate                      | insp/min | resp_rate              |
| 224689 | Respiratory Rate (spontaneous)        | insp/min | resp_rate              |
| 224690 | Respiratory Rate (Total)              | insp/min | resp_rate              |
| 223762 | Temperature Celsius                   | °C       | temp                   |
| 223761 | Temperature Fahrenheit                | °F       | temp                   |
| 224639 | Daily Weight                          | kg       | weight                 |
| 226512 | Admission Weight (Kg)                 | kg       | weight                 |
| 220546 | WBC                                   |          | white_blood_cell_count |

Table 2: Exact ITEMIDs extracted from CHARTEVENTS table in MIMIC-IV

have distinct ITEMID codes in the EHR. We use the groupings provided in [Wang et al. \(2020\)](#), given explicitly in Tab. 2 for our variables of interest. Note that we extracted weight from the charts over time, but treated it as static (not dynamic) due to the limited 48 hour window.

### D.3 Hyperparameters and Computational Hardware

For the full hyperparameter grid, see Table 3. The logistic regression model was allowed a larger hyperparameter grid to compensate for its reduced parameter size, attempting to give each model family roughly equal computation runtime. All experiments were run using 4 CPU cores and 16GB of memory on a high-performance computing

cluster. Only 30 minutes of walltime was allowed for each model fitting, and all completed within this limit.

| Hyperparameter                  | Logistic Regression          | MLP                      | Random Forest |
|---------------------------------|------------------------------|--------------------------|---------------|
| Inverse Regularization Strength | 10, 100, 1000, 10000, 100000 | –                        | –             |
| Max Iterations                  | 1000, 49, 7, 1               | 100, 200                 | –             |
| Rare Class Weight               | 1, 3, 9, 27                  | –                        | 1, 3, 9, 27   |
| Hidden Layer Sizes              | –                            | (64,), (64,64), (128,64) | –             |
| L2-Regularization Strenght      | –                            | 0.0001, 0.001            | –             |
| Learning Rate                   | –                            | 0.001, 0.0005            | –             |
| Max Depth                       | –                            | –                        | 4, 16, 64     |
| Minimum Examples in Leaf        | –                            | –                        | 4, 16, 64     |

Table 3: Hyperparameter grid used with MIMIC-IV experiments.

#### D.4 MIMIC-IV: Procedure for model selection and deployment-aware testing

We define at the outset some desired constraints via specific values of  $\alpha, \kappa$ , as well as a desired density over the fractional cost-parameter  $p(t)$  that is *potentially non-uniform*, but that satisfies  $\int_{t=a}^b p(t)dt = 1$  over a provided range  $[a, b] \subseteq [0, 1]$ . For example, we can define a desired distribution over the cost ratio  $C_0/C_1$  as in main paper, and map this to a density  $p(t)$  either explicitly (via change of variables) or implicitly (via sampling cost ratios and mapping each sample to  $t$ ). Note that while our main paper defined partial VOROS as an integral over a uniform  $t$  density, both VOROS and partial VOROS can account for a non-uniform  $p(t)$  within the integral naturally, as described in [Ratigan and Cowen \(2025\)](#).

**Development and model selection phase: Use train + validation data.** For each model family  $m$  in LR, RF, and MLP, we train a set  $\mathcal{H}_m$  of different candidates across a spectrum of hyperparameters designed to span under-fitting and over-fitting (see grid in Tab. 3), hopefully including some well-fitting model instances. Denote the union of all model configurations as  $\mathcal{H} = \mathcal{H}_{LR} \cup \mathcal{H}_{RF} \cup \mathcal{H}_{MLP}$ . Each element in  $\mathcal{H}$  results in a score-producing classifier with its own ROC curve on the validation set. Using all validation set ROC curves and provided  $\alpha, \kappa$  limits, we pick the single model-hyperparameter combination  $h \in \mathcal{H}$  that performs best using various selection strategies:

- maximizing partial VOROS across given range  $t \in [a, b]$  and density  $p(t)$ , accounting for  $\alpha, \kappa$
- maximizing total VOROS across given range  $t \in [a, b]$  and density  $p(t)$  (ignoring  $\alpha, \kappa$ )
- maximizing recall in feasible region defined by  $\alpha, \kappa$  (ignoring  $t$ )
- maximizing partial AUROC in feasible region defined by  $\alpha, \kappa$  (ignoring  $t$ )

We select a single winning configuration  $h^* \in \mathcal{H}$  for each strategy. Using this  $h^*$ , we can revisit each  $t$  value in the given range, and determine a specific binarization threshold  $\tau(t)$  that performs best at that  $t$ . Some cost-unaware strategies like maximizing recall will use the same  $\tau$  always, so  $\tau(t)$  is just a flat function.

**Test phase.** We wish to mimic authentic deployment of a real alert system for given limits of  $\alpha, \kappa$  and costs  $t \in [a, b]$  weighted by the given density  $p(t)$ . We will force each selected model on the test data to use its determined  $t$ -specific threshold  $\tau(t)$  to perform alerts, thus producing binary predictions (not scores) on the test set for a given  $t$ .

On the test set using each predetermined cost-aware threshold  $\tau(t)$ , we record the fpr, tpr location in ROC space as  $(h_{\tau(t)}, k_{\tau(t)})$ . We then report the expected cost over the provided range:

$$\begin{aligned}
\mathbb{E}_{t \sim p(t)}[\text{cost}] &= \int_{t=a}^b p(t) \text{cost}(h_{\tau(t)}, k_{\tau(t)}, t) dt \\
&= \int_{t=a}^b p(t) [th_{\tau(t)} + (1-t)(1-k_{\tau(t)})] dt
\end{aligned} \tag{1}$$

We can approximate this last integral over a sufficiently dense grid of  $t$  values numerically via the trapezoid approximation.

---

If we have the ability to sample instead of evaluate the explicit density function  $p(t)$ , we can approximate the expected cost on the test set as:

$$\approx \frac{1}{S} \sum_{s=1}^S \left[ t_s h_{\tau(t_s)} + (1 - t_s)(1 - k_{\tau(t_s)}) \right] \quad (2)$$

where the last formula uses an  $S$  sample Monte Carlo approximation, with each  $t_s$  drawn independently from  $p(t)$ .